Homework 1 Distributed Database Systems

Christoffer Brevik

October 22, 2024

Introduction

This document shows my process and answer for the problems given in the homework.

Database used in assignment

Below are the test data described as *Figure 5.3* in the assignment. The following tables are used as part of the solutions for the homeworks problems.

EMP

ENO	ENAME	TITLE
E1	J. Doe	Elect. Eng.
E2	M. Smith	Sys. Anal.
E3	A. Lee	Mech. Eng.
E4	J. Miller	Programmer
E5	B. Casey	Sys. Anal.
E6	L. Chu	Elect. Eng.
E7	R. Davis	Mech. Eng.
E8	J. Jones	Sys. Anal.

ASG

ENO	PNO	RESP	DUR
E1	P1	Manager	12
E2	P1	Analysist	24
E2	P2	Analysist	6
E3	P3	Consultant	10
E3	P4	Engineer	48
E4	P2	Programmer	18
E5	P2	Manager	24
E6	P4	Manager	48
E7	P3	Engineer	36
E8	P3	Manager	40

PROJ

PNO	PNAME	BUDGET	LOC
P1	Instrumentation	150,000	Montreal
P2	Database Develop.	135,000	New York

PNO	PNAME	BUDGET	LOC
P3	CAD/CAM	250,000	New York
P4	Maintenance	310,000	Paris

PAY

TITLE	SAL
Elect. Eng.	40,000
Sys. Anal.	34,000
Mech. Eng.	27,000
Programmer	24,000

1 Problem 5.2

Application 1

As all the sites contain one of the 5 different responsibility roles, we can divide the database into the following RESP values to facilitate this:

- RESP = 'Manager'
- RESP = 'Analasist'
- RESP = 'Consultant'
- RESP = 'Engineer'
- RESP = 'Programmer'

Application 2:

As the two sites that use this application contain employees with less than 20 months and more than 20 months assignments, respectfully, we can also separate the values to facilitate this:

- DUR <20
- DUR >= 20

Primary horizontal Fragmentation

By combining these predicates, we end up with the following Primary Horizontal Fragmentation (PHF):

- F1: (Manager, DUR <20)
- F2: (Manager, DUR >= 20)
- F3: (Analasist, DUR <20)
- F4: (Analasist, DUR >= 20)
- F5: (Consultant, DUR <20)
- F6: (Consultant, DUR >= 20)
- F7: (Engineer, DUR <20)
- F8: (Engineer, DUR >= 20)
- F9: (Programmer, DUR <20)
- F10: (Programmer, DUR >= 20)

Here it is important to note that both F6 and F10 would not actually contain any rows as none of the employees in the database match these criteria. Therefore we would normally not contain these fragments as they would serve no purpose if they don't contain any data.

2 Problem 5.8

Application 1:

We have the following application, hereby called q1

```
CREATE VIEW EMPVIEW     (ENO, ENAME, TITLE, PNO, RESP) =
        AS SELECT       EMP.ENO, EMP.ENAME, EMP.TITLE, ASG.PNO, ASG.RESP
        FROM            EMP, ASG
        WHERE           EMP.ENO=ASG.ENO AND ASG.DUR=24 AND EMP.TITLE="Programmer"
```

Application 2:

We also have this second application, hereby called q2

```
SELECT  ENO, DUR
FROM    ASG
```

Variables

From q_1 and q_2 we have the following variables:

$$\begin{aligned} &\text{EMP:} \quad A_1 = \text{ENO}, \quad A_2 = \text{ENAME}, \quad A_3 = \text{TITLE} \\ &\text{ASG:} \quad A_4 = \text{ENO}, \quad A_5 = \text{PNO}, \quad A_6 = \text{RESP}, \quad A_7 = \text{DUR} \end{aligned}$$

use(q_i, A_j) Matrix

We can with the information above create the use(q_i, A_j) matrix:

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
q_1	1	1	1	1	1	1	1
q_2	0	0	0	1	0	0	1

Frequency Matrix

We can also create a matrix fdisplaying the frequency of the two applications used by the three different sites:

	S_1	S_2	S_3
q_1	10	20	0
q_2	0	20	10

Calculating the affinity matrix

We can now create an affinity matrix. To save space, I will simplify my row and column calculation by showing my calculations in the following manner.

For $n, iin[1, 2, 3, 5, 6]$:

$$(A_n, A_i) = (10 + 20)_{q_1} = 30$$
$$(A_n, A_{4,7}) = (10 + 20)_{q_1} = 30$$

For 4 og 7:

$$(A_{4,7}, A_{4,7}) = (10 + 20)_{q_1} + (20 + 10)_{q_2} = 60$$

With these calculations I got the following matrix:

	A_1	A_2	A_3	A_4	A_5	A_6	A_7
A_1	30	30	30	30	30	30	30
A_2	30	30	30	30	30	30	30
A_3	30	30	30	30	30	30	30
A_4	30	30	30	60	30	30	60
A_5	30	30	30	30	30	30	30
A_6	30	30	30	30	30	30	30
A_7	30	30	30	60	30	30	60

As A_4 and A_7 are the largest values, we can shuffle the array to be

	A_4	A_7	A_1	A_2	A_3	A_5	A_6
A_4	60	60	30	30	30	30	30
A_7	60	60	30	30	30	30	30
A_1	30	30	30	30	30	30	30
A_2	30	30	30	30	30	30	30
A_3	30	30	30	30	30	30	30
A_5	30	30	30	30	30	30	30
A_6	30	30	30	30	30	30	30

Here we see that A_4 and A_7 are tightly connected. Therefore, it could be smart to perform vertical separation with these in mind. Importatnly, the $A_7 : ASG.DUR$ is dependent on the primary keys $A_4 : ASG.ENO$ and $A_5 : ASG.PNO$ I would therefore propose the following vertical fragmentation of ASG:

$$R_1 = (A_4, A_5, A_7), \quad R_2 = (A_4, A_5, A_6)$$

This would lead to the following matrices:

ASG_R1

ENO	PNO	DUR
E1	P1	12
E2	P1	24
E2	P2	6
E3	P3	10
E3	P4	48
E4	P2	18
E5	P2	24
E6	P4	48
E7	P3	36
E8	P3	40

ASG_R2

ENO	PNO	RESP
E1	P1	Manager
E2	P1	Analysist
E2	P2	Analysist
E3	P3	Consultant
E3	P4	Engineer
E4	P2	Programmer
E5	P2	Manager
E6	P4	Manager
E7	P3	Engineer6
E8	P3	Manager

3 Problem 5.17

Using the information from the affinity matrix in Problem 5.8 and the updated information, we will find a structure which will optimize speed and replication.

Fragmentation Strategy

Horizontal Fragmentation for ASG (Based on DUR = 24)

The predicate `DUR = 24` indicates that we can perform horizontal fragmentation of the ASG relation as follows:

- **ASG_F1**: Contains tuples where `DUR < 24`
- **ASG_F2**: Contains tuples where `DUR >= 24`

Horizontal Fragmentation for EMP (Based on TITLE = 'Programmer')

The condition `TITLE = 'Programmer'` suggests we can horizontally fragment the EMP relation:

- **EMP_F1**: Contains tuples where `TITLE = 'Programmer'`
- **EMP_F2**: Contains tuples where `TITLE` $\neq$ `'Programmer'`

Vertical Fragmentation for ASG (Based on Affinity Matrix)

From the affinity matrix in Problem 5.8, attributes `ASG.ENO` (A4) and `ASG.DUR` (A7) are frequently accessed together. Therefore, a vertical fragmentation of ASG makes sense:

- **R1**: `(ASG.ENO, ASG.PNO, ASG.DUR)`.
- **R2**: `(ASG.ENO, ASG.PNO, ASG.RESP)`.

Summary of fragments

Name	Values	Restriction
EMP_F1	ENO, ENAME, TITLE,	TITLE = "Programmer"
EMP_F2	ENO, ENAME, TITLE	TITLE $\neq$ "Programmer"
ASG_R1_F1	ENO, PNO, DUR	DUR < 24
ASG_R1_F2	ENO, PNO, DUR	DUR >= 24
ASG_R2	ENO, PNO, RESP	

Replication and Placement Strategy

From the assignment

- **Data Transfer Rates**: The transfer rate between Site 1 and Site 2 is half that between Site 2 and Site 3, making replication between Sites 2 and 3 more cost-effective.
- **Query Access Patterns**: 60% of query `q1` accesses are updates to the `PNO` and `RESP` fields of EMP, so fragments containing these attributes need careful replication to ensure consistency.

Replication of EMP Fragments

- **EMP_F1**: Placed at Site 2, as this site sees the most activity for query `q1` (20 updates). It is also replicated at Site 3 for consistency, due to high traffic between Sites 2 and 3.
- **EMP_F2**: Placed at Site 3, since it has lower update activity.

Replication of ASG Fragments

- **ASG_R1_F1** (with `DUR < 24`): Placed at Site 1, where minimal query activity for `q1` occurs.
- **ASG_R1_F2** (with `DUR >= 24`): Placed at Site 2, where query `q1` focuses on employees with `DUR = 24`. It is replicated to Site 3 for consistency.
- **ASG_R2**: Placed at Site 2 and replicated to Site 3, similar to **ASG_R1_F2**.

Final Placement and Replication Summary

Site	Fragments	Placement Details
Site 1	ASG_R1_F1	Tuples with `DUR < 24`
Site 2	EMP_F1	Tuples where `TITLE = 'Programmer'`
	ASG_R1_F2	Tuples with `DUR >= 24`
	ASG_R2	Tuples with `ENO, PNO, RESP`
Site 3	EMP_F2	Tuples where TITLE $\neq$ 'Programmer'
	EMP_F1	Replicated from Site 2
	ASG_R1_F2	Replicated from Site 2
	ASG_R2	Replicated from Site 2